Supporting End-to-End QoS in DiffServ/MPLS Networks

Ji-Feng Chiu, *Zuo-Po Huang, *Chi-Wen Lo, *Wen-Shyang Hwang and Ce-Kuen Shieh
Department of Electrical Engineering,
National Cheng Kung University, Taiwan R.O.C.
*Department of Electrical Engineering,
National Kaohsiung University of Applied Sciences, Taiwan R.O.C.
E-mail: *gary@hpds.ee.ncku.edu.tw, {adrian, ken}@wshlab2.ee.kuas.edu.tw*
wshwang@mail.ee.kuas.edu.tw, shieh@eembox.ee.ncku.edu.tw

Abstract — **In MPLS networks, the higher-priority LSP (Label Switching Path) will preempt the resource of lower-priority LSP when its bandwidth resource is limited. At the time, the lower-priority LSP is destroyed, and its bandwidth resource is released to the higher-priority LSP. The destroyed path has to be rerouted by selecting another LSP; the new LSP maybe quickly suffer another bandwidth resource preemption again. If this scenario occurs frequently, routers would be subjected to the superfluous overload, and the quality of delivering flow can not be guaranteed. In paper [15], we had proposed a new policy to avoid the preemption because of flow priority and load balancing in the MPLS networks. In this paper, the policy supports the end-to-end QoS in DiffServ-aware MPLS networks will be discussed. Moreover, many simulations are done to compare the QoS of all service levels in DiffServ which Constraint-based Routed (CR) scheme includes the new policy in paper [15]. The simulation results indicate that the adding our policy to CR is better then the traditional CR.**

Keywords: End-to-End QoS, DiffServ, MPLS and Traffic Engineering

I. INTRODUCTION

The traditional Internet only provides best effort service; stations transmit packets as quickly as possible. There is no guarantee as to timeless or actual delivery. In the past several years, new types of Internet applications that require performance guarantees beyond the best effort service have emerged. These applications do not have any strict service level requirements; however, there are mission critical applications. The Internet Engineering Task Force (IETF) has proposed many service models and protocols for providing QoS in the Internet, such as Integrated Service (IntServ), Differentiated Service (DiffServ) and Multi-Protocol Label Switching (MPLS). IntServ is an architecture that requires the per-flow traffic handling at every hop along the application's end-to-end path, and the explicit signaling of each flow's requirements using a signaling protocol like Resource reSerVation Protocol (RSVP) [1][2]. IntServ suffers from lack of scalability due to the scalability problems with the standard RSVP signaling protocol. Therefore, the DiffServ architecture was proposed. The differentiated service architecture is based on a simple model where traffic entering a network is classified and possibly conditioned at the boundaries of the network, and assigned to different behavior aggregates that are a collection of packets with common characteristics. Each behavior aggregate is identified by a single DSCP (Differentiated Services CodePoint) [3][4]. Within the core of the network, packets are forwarded according to the Per-Hop Behavior (PHB) associated with the DSCP. There are three defined PHBs: (i) Best Effort (BE), (ii) Assured Forwarding (AF) and (iii) Expedited Forwarding (EF). The AF PHB supports more flexible and dynamic sharing of network resourced by soft bandwidth and loss guarantees appropriate for burst traffic [5]. The EF PHB requests every router along the path to always service EF packets at any rate as fast as the rate at which EF packets arrive [6]. MPLS is a technology that integrates label-switching forwarding paradigm with network layer routing, such as ATM (Asynchronous Transfer Mode) or frame relay. It offers an aggregated data path for all services, while allowing combinations of control plane scheme within the same backbone to provide multiple logical service networks. With MPLS, it is possible to set up routes on the basis of the individual flows, with two different flows between the same end-points perhaps following different routers. Further, when congestion occurs, LSPs can be rerouted automatically [7-8]. The most important application of MPLS is in traffic engineering [9] [10] [11] [12]. Traffic Engineering is the process of routing traffic in order to balance the network. Redirecting packets to other than the best shortest path calculated by routing protocols usually does this. DiffServ and MPLS are now viewed as complementary in the pursuit of end-to-end QoS provisioning [13] [14]. Consequently, The DiffServ adds MPLS in order to provide the QoS guarantees for customer, the efficient network resource requirements by network providers, and the reliability and adaptation of node and link failures. DiffServ provides the scalable edge-to-edge QoS, while MPLS performs traffic engineering to evenly distribute traffic load on available links, fast rerouting to route around node, and link failures in order to end-to-end QoS for customer's applications.

In MPLS networks, the higher-priority LSP (Label Switching Path) will preempt the resource of lower-priority

0-7803-7661-7/03/$17.00©2003 IEEE

LSP when bandwidth resource is restrained. The LSP preemption introduces a setup and holding priority. When preemption occurred, the lower-priority LSP will be destroyed, and its bandwidth resource is released. The higher-priority LSP obtains the bandwidth resource to establish its path. While the lower-priority LSP release bandwidth, it has to be rerouted by selecting another LSP, but the LSP cannot ensure whether its bandwidth resources will be preempted again or not. If this situation occurred frequently, routers would have superfluous overload; quality of flow delivering is disappointment. In the [15], we had proposed a new policy in order to avoid preemption for every priority flow and load balancing in the MPLS networks. It also discussed the performance aim at MPLS networks. In this paper, we will discuss its feasibility that the policy supports end-to-end QoS in DiffServ-aware MPLS networks. Simulations will compare QoS of each service level over DiffServ whether constraint-based routed (CR) scheme add our policy.

The rest of this paper is organized as follows: Section II introduces the related work of DiffServ, MPLS and TE. Section III describes the policy proposed in the paper. Section IV provides simulation results. Finally, a conclusion and future work are presented.

II. RELATED WORKS

We will not include related studies for DiffServ [3-6], MPLS [7-8] [20-23] and traffic engineering [9-12] [21] [24-25] in this section because of page limitation. So far several researchers have proposed many schemes in DiffServ-aware MPLS networks. Paper [16] showed how MPLS combined with differentiated services and constraint-based routing forms a simple and efficient Internet model capable of providing applications with differential QoS. No per-flow state information is required leading to increased scalability. They also proposed how this service architecture can interoperate with neighboring regions supporting IntServ and DiffServ QoS mechanisms. Paper [17] combined DiffServ technology with traffic engineering over MPLS to offer an adaptive mechanism that is capable of routing high priority IP traffic over multiple parallel paths to meet delay time constraints. They propose a probe packet method to collect delay measurements along several parallel paths. They use them in an end-to-end delay predictor that outputs a quick current estimate of the end-to-end delay. Paper [18] proposed network structure and the algorithm offer a solution that dynamically determines QoS-constrained routes with a number of demands and routes traffic within the network so that the demands are carried with the requisite QoS while fully utilizing network resources. Applying the central resource manager they remove the complexity of finding QoS routes at the core of the network. Finally, by using the modified version of Dijkstra's algorithm they provide a solution for dynamical determining QoS-constrained routes while balancing the load on the network. Paper [19] proposed a per-class TE scheme that enhances E-LSP. The scheme is carefully analyzed against TE requirements. However, they have not proposed any actual mechanisms that can ensure bandwidth resources preempted again for every service lever in order to obtain end-to-end QoS when it preempted.

III. PREEMPTION AVOIDANCE AND LOAD BALANCING POLICY

We assume that our policy operates in a MPLS network supporting traffic engineering and these LSPs are established using CR-LDP that is from ingress router to egress router. In these LSPs setup procedure, our policy is used for each link of each LSP to calculate *Preemption Probability*.

```
1.  Using CR-LDP find those LSPs that are from ingress router to egress router
2.  If LSPs exist and remaining bandwidth $\geq$ requisition bandwidth

    For each links of each LSPs compute its preemption probability
        $\left( \sum_{i=0}^{i<flow} P_i \Big/ N_p \right) \times \left( \sum_{j=b}^{j<Total} B_j \Big/ B_{Total} \right)$
      Select maximum preemption probability for the LSP in each links of the LSP
      Select minimum preemption probability for the flow in each LSPs

    elseif LSPs exist and requisition bandwidth > remaining bandwidth
      It preempt other LSPs to service the flow
    elseif LSPs not exist or it cannot preempt other LSPs
      Rejecting request
```

Fig.1. Our policy for preemption avoidance and load balancing

The flow of preempting probability ($P_{(flow)}$) is defined as equation (1), which means the probability of this LSP be preempted by other higher-priority LSPs. It combines priority level and requisition bandwidth both. In this case, we assume that the arrival rate and requisition bandwidth for each level LSP equally. All LSP with priority greater than this LSP and requires bandwidth more than the available bandwidth of the link can be expressed as

$$P_{(flow)} = \left(\sum_{i=0}^{i<flow} P_i \Big/ N_p \right) \times \left(\sum_{j=b}^{j<Total} B_j \Big/ B_{Total} \right) \qquad (1)$$

Where N_p is level of all priority, P_i is the priority of the flow i_{th}, B_{total} is the link bandwidth, and b is remaining bandwidth of the link.

After each links of all LSPs are calculated, the LSP selects the maximum preemption probability to stand for it. Finally, our policy selects the LSP to deliver packets with the minimum preemption probability in all LSPs, called preemption avoidance, are shown in Figure 1.

IV. SIMULATION RESULTS

In this section, we show the simulation results in the previous section. We used the network simulator NS-2 [26] with new modules supporting our policy to CR described earlier. The network topology used in our simulation is shown in Figures 2. The link bandwidth between N0 and N1, N0 and N2, N2 and N3, N3 and N5, N3 and N6, N5 and N6, are set to 45Mbps. Propagation delay of the links are 0.1ms. Another links are set to 25Mbps. Propagation delay of the links are 0.05ms. S1, S2, and S3 send BE (20Mbps), AF (15Mbps), EF (25Mbps) class data to R1, R2 and R3 respectively, and S4 send scenario a (EF), scenario b (AF), and scenario c (BE), data (20Mbps) to R1. The order of active time is S1, S2, S3, and S4. The simulation cases are designed to evaluate the effectiveness of the proposed policy for various service classes, and are summarized in Figures 3. In the simulation, we measure the throughput, delay and jitter of each services class every 0.1ms.

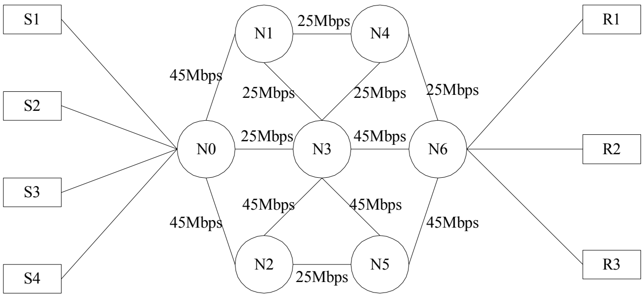

Fig.2 Simulation topology

A. S1 (BE class)

Figures 4, 5, and 6 show that S1 (BE) traffic performs independently under different schemes. In the case 1a, 1b, and 1c, S1 selects LSP 0-3-6 to delivery packets. We can see results that preemption occurred when the AF and EF traffic are active and it selects another LSP 0-1-4-6, LSP 0-2-5-6, and LSP 0-1-3-6 for S2 (AF), S3 (EF), and S4 (EF1a/AF1b) respectively. Therefore, the throughput, delay, and jitter became bad. The proposed scheme avoided occurring preemption; S1 selects LSP 0-3-6, S2 selects LSP 0-1-4-6, S3 selects LSP 0-2-3-5-6, and S4 selects 0-1-3-6. Although these flows do not select shortest path to delivery packets, they still reach expectable QoS.

Case 1a, 1b, 1c	Traditional CR without our policy S4 is EF, AF, and BE in case 1a, 1b, and 1c respectively
Case 2a, 2b, 2c	Traditional CR adds our policy S4 is EF, AF, and BE in case 2a, 2b, and 2c respectively

Fig.3 Simulation cases

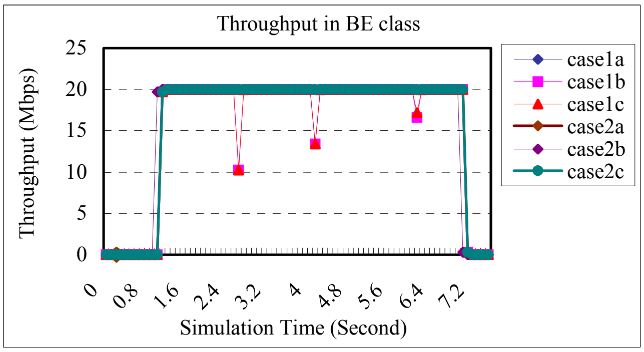

Fig.4 Throughput of S1

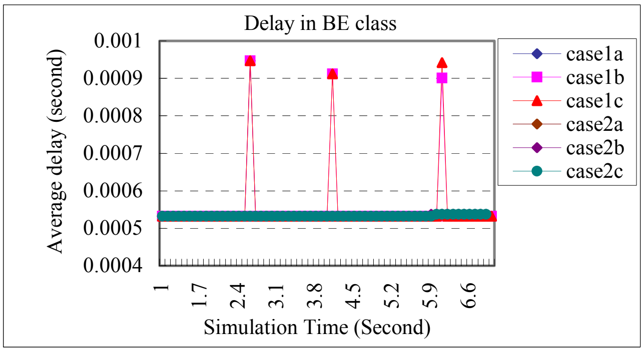

Fig.5 Delay of S1

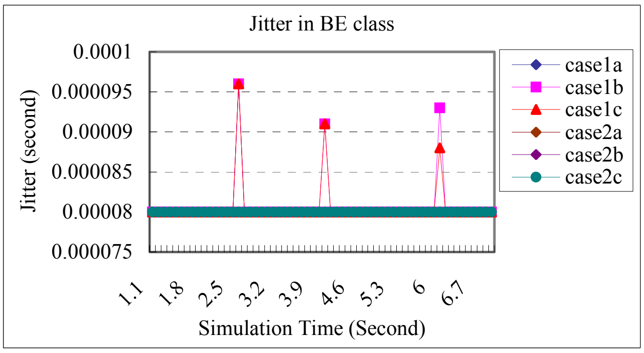

Fig.6 Jitter of S1

B. S2 (AF class)

Figures 7, 8, and 9 show that performance of S2 (AF) traffic. We see from these Figures that throughput, delay, and jitter obtain to improve in our scheme clearly.

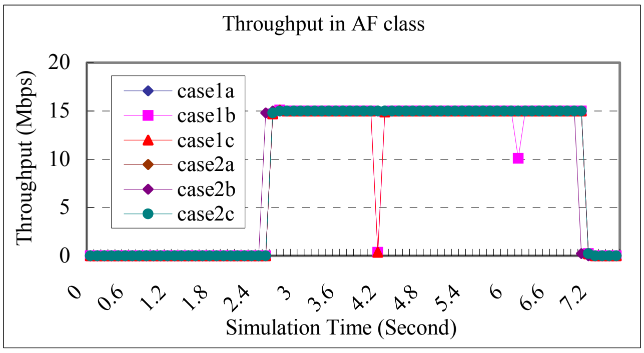

Fig.7 Throughput of S2

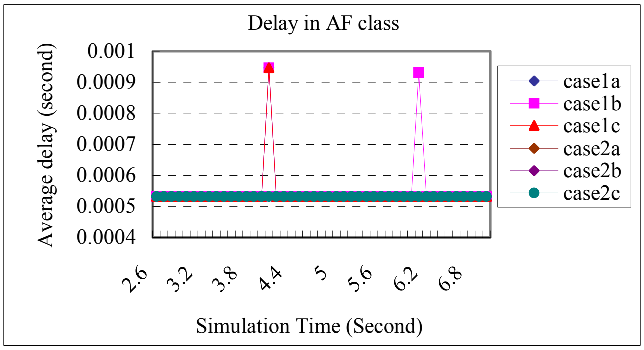

Fig.8 Delay of S2

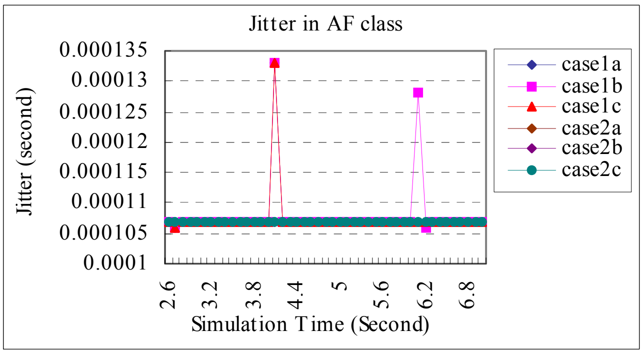

Fig.9 Jitter of S2

C. S3 (EF class)

Figures 10, 11, and 12 show throughput, delay, and jitter for S1 (BE) traffic. Although these flows do not select shortest path to delivery packets, they still reach expectable performance and avoid preemption in our scheme.

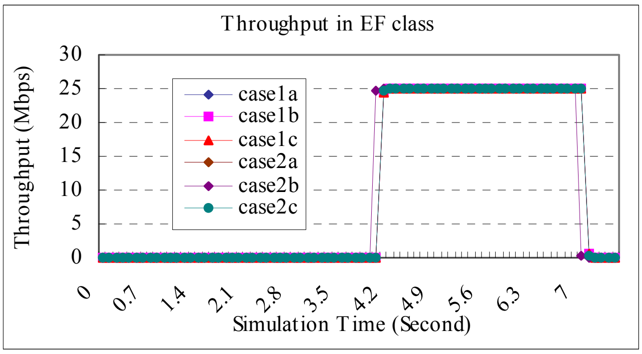

Fig.10 Throughput of S3

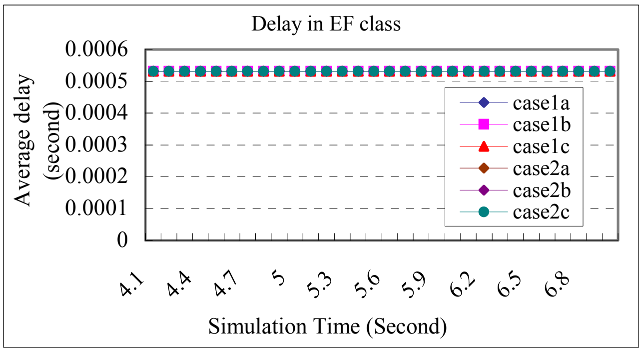

Fig.11 Delay of S3

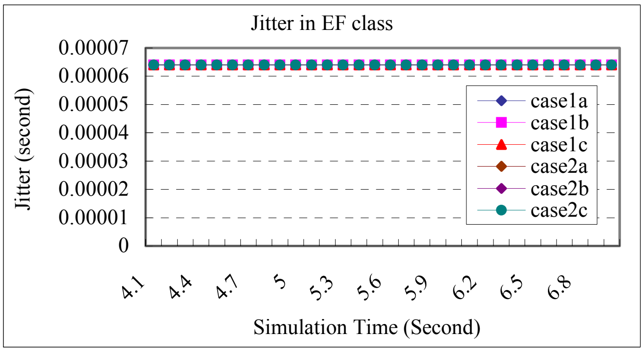

Fig.12 Jitter of S3

D. S4 (EF/AF/BE class)

Performance is shown in Figures 13, 14, and 15 for S4 (EF1, 2a /AF1, 2b/ BE1, 2c). In case1a, 1b, and 1c, S4 influences AF and BE service traffic to cause preemption when it is EF service class. The AF and BE service traffic reroutes by selection LSP. When it becomes AF service class, it influences BE service traffic. When it is BE service class, its performance has not influenced because it is active last one.

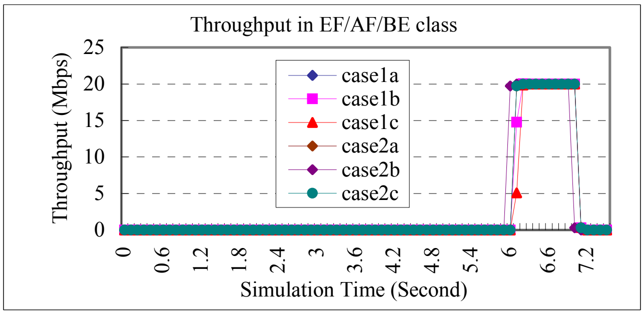

Fig.13 Throughput of S4

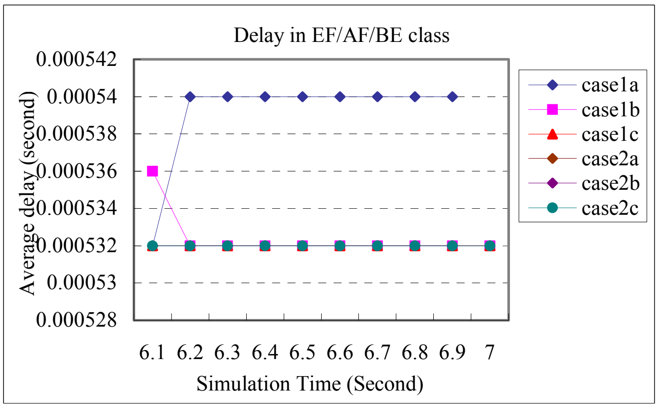

Fig.14 Delay of S4

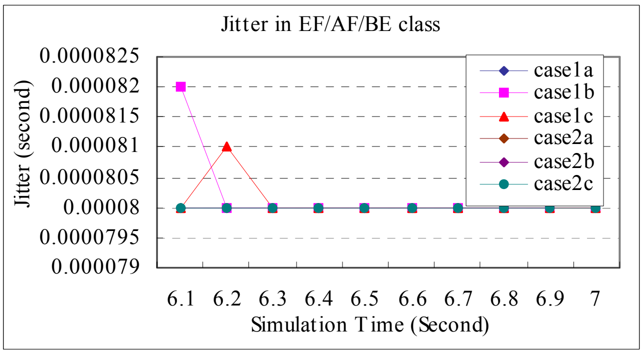

Fig.15 Jitter of S4

V. CONCLUSION AND FUTURE WORK

In this paper, we discuss a new policy in order to avoid preemption for every priority flow and load balancing in the MPLS networks that its feasibility that the policy supports end-to-end QoS in DiffServ-aware MPLS networks. The results from the simulation indicated that adding our policy to CR with comparable QoS of each service level is a better to traditional CR. Although higher-level flows did not select shortest path to delivery packets, they still reach expectable performance and avoid preemption.

In the future, we will plan to discuss more real distribution for priority-level and requisition bandwidth and another important advantage is MPLS-based traffic engineering by MPLS to DiffServ. Similar experiments can be done to highlight these advantages.

References

[1] Braden, R., Clark, D. and Shenker, S., "Integrated Services in the Internet Architecture: an Overview", Internet RFC 1633, June 1994

[2] R.Barden, Ed, “Resource ReSerVation Protocol (RSVP) -Version 1 Functional Specification,“ Internet RFC2205, September 1997.

[3] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss, “An Architecture for Differentiated Services,” RFC 2475, December 1998.

[4] K. Nichols, S. Blake, F. Baker and D. Black, “Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers,” RFC 2474, December 1998.

[5] Heinanen, J., Baker, F., Weiss, W. and J. Wroclawski, "Assured Forwarding PHB Group", RFC 2597, June 1999.

[6] Jacobson, V., Nichols, K. and K. Poduri, "An Expedited Forwarding PHB", RFC 2598, June 1999.

[7] E.Rosen, A.Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001

[8] T.Li, “MPLS and the evolving Internet architecture”, IEEE Communications Magazine, Volume: 37 Issue: 12 pp.38-41, Dec. 1999

[9] C.Srinivasan, A.Viswanathan, and T.Nadeau, "MPLS Traffic Engineering Management Information Base ", Internet Draft, January 2002

[10] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell and J. McManus, “Requirements for Traffic Engineering Over MPLS”, RFC 2702, September 1999

[11] X.Xiao, A.Hannan, B.Bailey, and L.M.Ni, “Traffic Engineering with MPLS in the Internet”, IEEE Network, Volume: 14 Issue: 2 pp.28-33, March-April 2000

[12] D.O.Awduche, “MPLS and traffic engineering in IP networks”, IEEE Communications Magazine, Volume: 37 Issue: 12 pp.42-47, Dec. 1999

[13] Le Faucheur et al., “MPLS Support of Differentiated Services”, RFC 3270, May 2002

[14] Li T., Rekhter Y., “A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)”, RFC2430, October 1998

[15] J.F Chiu et al., “A New Preemption Avoidance and Load Balancing Policy for Traffic Engineering in MPLS Networks”, Accepted by IASTED CCN 2002.

[16] N.Rouhana and E.Horlait, **“**Differentiated services and integrated services use of MPLS**”,** ISCC 2000, pp.194-199

[17] T.Saad, T.Yang, D.Makrakis and V.Groza, “DiffServ-enabled adaptive traffic engineering over MPLS”, ICII 2001, Volume: 2pp.128–133, 2001

[18] A.Chpenst and T.Curran, “Optimization of QoS traffic with a number of constraints in DiffServ/MPLS networks”, http://telecoms.eeng.dcu.ie/symposium/papers/D1.pdf

[19] M.Moh, B.Wei and J.H Zhu, “Supporting differentiated services with per-class traffic engineering in MPLS”, International Conference on Computer Communications and Networks 2001, pp.354-360, 2001

[20] B.Jamousi et. al., ”Constraint-based LSP setup using LDP”, RFC 3212, January 2002

[21] M. Kodialam and T.V.Lakshman, “Minimum Interference Routing with Applications to MPLS Traffic Engineering”, INFOCOM 2000, Volume: 2 pp.884–893, 2000

[22] P.Ashwood-Smith, B.Jamoussi, D.Fedyk, and D.Skalecki, “Improving Topology Data Base Accuracy with LSP Feedback in CR-LDP”, draft-ietf-mpls-te-feed-04.txt, May 2002

[23] A.Iwata, N.Fujita and G.Ash, “Crankback Routing Extensions for CR-LDP, draft-fujita-mpls-crldp-crankback-01.txt, July 2000

[24] A.Elwalid, C.Jin, S.Low and I.Widjaja, “MATE: MPLS adaptive traffic engineering”, INFOCOM 2001, Volume: 3 pp.1300-1309, 2001

[25] J.C.de Oliveira, C.Scoglio, I.F.Akyildiz and G.Uhl, “A New Preemption Policy for DiffServ-Aware Traffic Engineering to Minimize Rerouting”, INFOCOM 2002, June 2002.

[26] NS-2, http://www.isi.edu/nsnam/ns/